УДК 004.89
DOI https://doi.org/10.32782/2663-5941/2023.5/33

***Пулеко І.В.***
Державний університет «Житомирська політехніка»

***Побережна М.О.***
Державний університет «Житомирська політехніка»

***Кравченко С.М.***
Державний університет «Житомирська політехніка»

***Дмитренко І.А.***
Державний університет «Житомирська політехніка»

# РІШЕННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ МЕДИЧНИХ ДАНИХ НА ОСНОВІ ПОЄДНАННЯ СЛУЖБИ МАШИННОГО НАВЧАННЯ AZURE ТА РОЗРОБЛЕНОГО ВЕБ-ДОДАТКУ

*На даний момент машинне навчання (МН) є одним з найактуальніших напрямків у розробці програмного забезпечення. Особливо широко індустрія охорони здоров'я використовує МН для створення веб-додатків, які спроможні передбачати ризики та виявляти захворювання на ранніх стадіях, з метою поліпшення обслуговування пацієнтів. Описаний у роботі процес розробки такого веб-додатку включає три етапи: створення моделі машинного навчання в Azure, розробку самого веб-додатку та інтеграцію розробленої моделі в нього, а також розгортання веб-сервісу з використанням Azure. Для першого етапу використано Azure Machine Learning Designer, який надає графічне середовище для створення моделей МН. У статті детально описані кроки створення моделі, такі як створення конвеєра, імпорт даних, підготовка даних, навчання моделі з використанням обраного алгоритму, а також оцінка та аналіз отриманої моделі. У розробці використана модель дерева рішень, яка має деревоподібну структуру: внутрішні вузли представляють характеристики даних, гілки відображають правила прийняття рішень, а кінцеві вузли представляють кінцеві результати. Ця модель добре підходить для вирішення задач класифікації, оскільки цільова змінна є дискретною. Після успішного створення та тестування моделі реалізований перехід до наступного етапу – створення веб-додатку з використанням технологій Vue.js, CSS та Node.js. Головна мета цього додатку – надання лікарям можливості швидко діагностувати пацієнтів, а іншим користувачам – доступу до результатів тестів. Для цього створено інтерфейс, який дозволить вводити вхідні дані та отримувати прогнозований результат. Наступним реалізованим етапом є розгортання моделей машинного навчання. Це означає впровадження моделей у виробниче середовище, де вони можуть використовуватися в реальному часі. Після розгортання моделі перевірено її ефективність з використанням тестових даних у форматі JSON, щоб переконатися, що веб-сервіс повертає очікувані результати.*

***Ключові слова:*** *аналіз даних, модель класифікації, машинне навчання, штучний інтелект, технології програмування, програмне забезпечення.*

**Постановка проблеми.** Сьогодні Інтернет став незамінним у нашому повсякденному житті. Належне використання Інтернету робить наше життя легким, швидким і простим. Інтернет допомагає нам фактами та цифрами, інформацією та знаннями для особистого, соціального та економічного розвитку. У зв'язку з розвитком інтернет-павутини веб-додатки зазнали чимало змін за останні роки. Розробка веб-додатків почала зростати швидше, ніж передбачалося. Потреба користувачів у отриманні розширеного досвіду та рідкісного контенту з роками багаторазово зросла. Це дозволяє нам зробити висновок, що користувач шукає інтелектуальні та інноваційні веб-додатки, які не тільки надають йому, керований даними, контент, але й дарують йому нестандартні ідеї. Таким чином, є лише одна річ, яка дозволяє задовольнити ці потреби, і це штучний інтелект. Актуальність впровадження штучного інтелекту (ШІ), зокрема машинного навчання (МН) у веб-додаток зростає з кожним роком. Таке рішення не тільки задовольняє потреби

користувачів та покращує кожну сферу життя людства, але й рятує життя. МН широко застосовується в медицині, що допомагає розроблювати нові ліки, оброблювати величезну кількість даних, а найголовніше це діагностувати захворювання на ранніх етапах, що допомагає уникнути летального результату. Впровадження МН у різні додатки дає змогу користувачам краще слідкувати за своїм здоров'ям, а лікарям швидше надавати допомогу пацієнтам.

Подальше дослідження потребує вирішення задачі побудови моделі для аналізу та прогнозування наявності хвороби пацієнта, використовуючи алгоритми класифікації із застосуванням можливостей Machine Learning на хмарній платформі Azure та інтегрування даної моделі у розроблений веб-додаток.

**Аналіз останніх досліджень і публікацій.** Дослідження з питань штучного інтелекту, зокрема машинного навчання, достатньо активно проводяться у колі науковців та стосуються багатьох сфер життєдіяльності людини. Окремою групою можна розглядати й наукові роботи медичного спрямування, де в основі лежить збір, аналіз та класифікація медичних даних, зокрема із використанням ШІ.

Загальний короткий нарис з поточної теми наведено у тезах авторів Пулеко І.В. та Побережна М.О. [1] У роботі зазначена проблематика, актуальність та наведений загальний план дослідження у рамках даної теми.

У публікації авторів Гадецька З. та Меркотан М. [2] розглядаються актуальні аспекти аналізу та прогнозування рівня захворюваності на COVID-19 у країнах Європейського регіону. Дослідження фокусується на трьох країнах із різним рівнем економічного розвитку: Німеччина, Іспанія та Україна. Для аналізу використовуються статистичні дані з ресурсу Our World in Data, спрямованого на вирішення глобальних проблем. Для прогнозування рівня захворюваності на COVID-19 застосовується модель ARIMA, що відноситься до класу авторегресії інтегрованого ковзкого середнього. Для підбору оптимальних коефіцієнтів ARIMA-моделі використовуються програмні продукти EViews та додаткові засоби, такі як набір надбудов Microsoft Excel, які призначені для проведення економетричного аналізу часових рядів. У даній роботі детально розглядається методологія збору та обробки даних, вибір моделі для прогнозування, а також інструменти, що використовуються для налагодження та аналізу моделі.

Автор Скопівський С.Я. [3] в своїй роботі сконцентрувався на порівнянні ефективності різних керованих алгоритмів машинного навчання для прогнозування захворювань. В рамках дослідження розглядається аналіз різноманітних підходів до прогнозування захворювань COVID-19. Зокрема, автор досліджує використання методів машинного навчання на статистичних даних, що стосуються поширення COVID-19, його взаємодії з іншими захворюваннями. У даній роботі надається детальний огляд різних підходів до прогнозування захворювання та розглядається застосування методів машинного навчання для аналізу статистичних даних. Автор аналізує, як COVID-19 впливає на статистику розповсюдження і взаємодію з іншими захворюваннями.

У статті від автора О.В. Гойка [4] акцентується увага на проблемі наукового аналізу медичних даних, використовуючи передові технології. Подані конкретні рекомендації щодо вибору методу обробки та програмного забезпечення, а також підготовки даних для комп'ютерного аналізу та обробки медичних спостережень. У даній статті звертається увага на проблеми, пов'язані з науковим аналізом медичних даних, і висвітлюється важливість використання сучасних технологій. Надаються конкретні поради щодо вибору оптимальних методів обробки та програмного забезпечення, а також підготовки даних для проведення комп'ютерного аналізу та обробки медичних спостережень.

У своєму дослідженні авторська група в складі Івченко В.К., Івченко А.В., Гальченко В.Я. та ін. [5] здійснила побудову прогностичної моделі, яка базується на аналізі біохімічних досліджень конкретних показників, що стосуються патогенезу ускладнень. В процесі розробки моделі використано метод лінійного непараметричного дискримінантного аналізу, який дозволив об'єднати клінічні показники та метаболічні порушення в єдину математичну модель для передбачення результатів лікування. Для досягнення прогнозних цілей стосовно кожного окремого пацієнта виконується розрахунок трьох функцій, які виконують класифікацію. Розрахунки базуються на центрованих і нормованих значеннях ознак. Пацієнта відносять до певного класу в залежності від того, яка з дискримінантних функцій набуває максимального значення. Створена програма визначає ймовірність або рівень ризику розвитку ускладнень при лікуванні переломів довгих кісток у хворих на цукровий діабет.

У статті від авторів Півошенко В.В., Кулик М.С. та ін. [6] детально розглядається сучасний метод машинного навчання, відомий як навчання з підкріпленням. Основний акцент зроблений на тому, що цей підхід має деякі ключові переваги. Зокрема, автори звертають увагу на можливість розвитку бота «з нуля» завдяки збалансованому поєднанню режимів «дослідження» та «застосування», а також вивченню стратегій, які дозволяють приймати найкращі рішення на основі компромісу між експлорацією та здобутками. У цій роботі автори представляють математичний апарат навчання з підкріпленням, в якому використовується метод безмодельного Q-навчання. Також надається огляд практичних аспектів застосування цього методу і стратегії навчання бота у штучному середовищі.

**Метою дослідження** є рішення задачі класифікації медичних даних на основі поєднання служби машинного навчання Azure та розробленого вебдодатку. У даній статті буде описана модель класифікації з використанням конструктора машинного навчання Azure. Цей етап буде включати наступні кроки:

– Попередній аналіз даних для знаходження закономірностей між різними функціями;

– Імпорт даних та їх аналіз за допомогою функцій конструктора Azure;

– Підготовка даних для навчання;

– Поділ даних на навчальний і тестувальний набори;

– Розробка та навчання моделі;

– Оцінка якості моделі;

– Розгортання моделі як кінцевого продукту.

**Виклад основного матеріалу дослідження**

**Аналіз вхідних даних**

Початкові дані були завантажені із репозиторію машинного навчання Каліфорнійського університету в Ірвіні ( UCI Machine Learning Repository). Репозиторій машинного навчання Каліфорнійського університету в Ірвіні є набір баз даних, теорій предметної області та генераторів даних, які використовуються спільнотою машинного навчання для емпіричного аналізу алгоритмів машинного навчання.

Репозиторій містить близько 600 наборів даних з різних предметів, включаючи інформатику, науки про життя, фізичні науки, бізнес, соціальні науки, ігри та багато іншого. Ми можемо шукати дані на основі характеристик (наприклад, табличних, часових рядів, послідовних, текстових) або пов'язаних завдань, для яких призначені дані (класифікація, регресія, кластеризація).

| | Age | Sex | CP | Restbps | Chol | BS | Restecg | MaxHR | Exang | Oldpeak | Slope | CA | Thal | Target |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 53 | 1 | 0 | 140 | 203 | 1 | 0 | 155 | 1 | 3.1 | 0 | 0 | 3 | 0 |
| 1 | 70 | 1 | 0 | 145 | 174 | 0 | 1 | 125 | 1 | 2.6 | 0 | 0 | 3 | 0 |
| 2 | 61 | 1 | 0 | 148 | 203 | 0 | 1 | 161 | 0 | 0.0 | 2 | 1 | 3 | 0 |
| 3 | 62 | 0 | 0 | 138 | 294 | 1 | 1 | 106 | 0 | 1.9 | 1 | 3 | 2 | 0 |
| 4 | 58 | 0 | 0 | 100 | 248 | 0 | 0 | 122 | 0 | 1.0 | 1 | 0 | 2 | 1 |

**Рис. 1. Перегляд набору даних**

Для навчання моделі класифікації необхідний набір даних, що включає

Стовпець «Age»: Вік пацієнта.

Стовпець «Sex»: Стать пацієнта (1 – чоловік, 2 – жінка).

Стовпець «ChestPainType»: Тип болю в грудях (1 – типова стенокардія, 2 – атипова стенокардія, 3 – неангінозний біль, 4 – симптоми відсутні).

Стовпець «RestingBlood»: Артеріальний тиск у стані спокою.

Стовпець «Chol»: Показник холестерину.

Стовпець «BloodSugar»: Цукор в крові натще > 120 мг/дл (1 – істинно, 0 – хибно).

Стовпець «Electrocardiographic»: Результати електрокардіографії в стані спокою (0 – нормальне, 1 – має аномалії хвилі ST-T, 2 – показ імовірної або певної гіпертрофії лівого шлуночка за критеріями Естеса).

Стовпець «MaxHeartRate»: Досягнутий максимальний пульс.

Стовпець «ExerciseAngina»: Стенокардія, спричинена фізичними вправами (1 – так, 0 – ні).

Стовпець «Oldpeak»: Депресія ST, спричинена фізичними вправами відносно відпочинку.

Стовпець «Slope»: Нахил піку вправи сегмента ST (1 – підвищення, 2 – плоский, 3 – спад).

Стовпець «NumOfMajorVessels»: Кількість основних судин (0–3), пофарбованих флюороскопією.

Стовпець «Thal»: Результати сканування товщини серця (3 – нормальний, 6 – виправлений дефект, 7 – оборотний дефект).

Стовпець «Target»: Результат діагностики серцевих захворювань (0 – від сутнє, 1 – наявне).

Для початку проаналізуємо загальні характеристики в наборі даних. Для початку розглянемо тип кожної змінної.

```
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Age       1024 non-null   int64
 1   Sex       1024 non-null   int64
 2   CP        1024 non-null   int64
 3   Restbps   1024 non-null   int64
 4   Chol      1024 non-null   int64
 5   BS        1024 non-null   int64
 6   Restecg   1024 non-null   int64
 7   MaxHR     1024 non-null   int64
 8   Exang     1024 non-null   int64
 9   Oldpeak   1024 non-null   float64
 10  Slope     1024 non-null   int64
 11  CA        1024 non-null   int64
 12  Thal      1024 non-null   int64
 13  Target    1024 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.1 KB
```

**Рис. 2. Перегляд типу кожного значення**

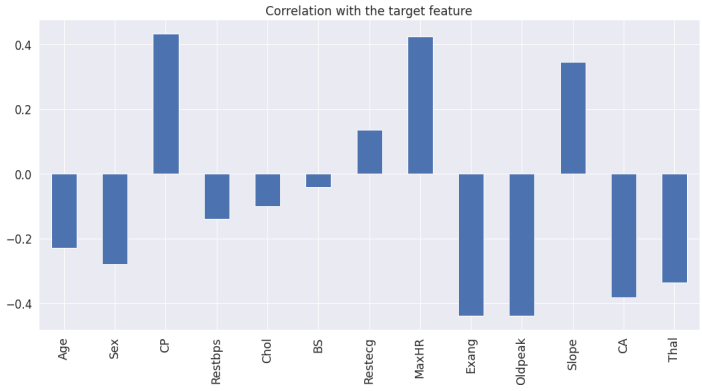


**Рис. 3. Результат кореляції цільової змінної**

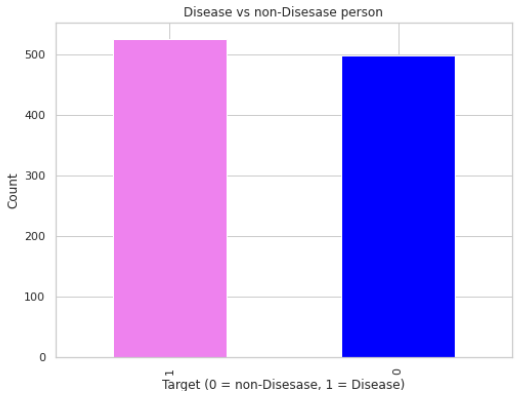


**Рис. 4. Кількість здорових і хворих людей**

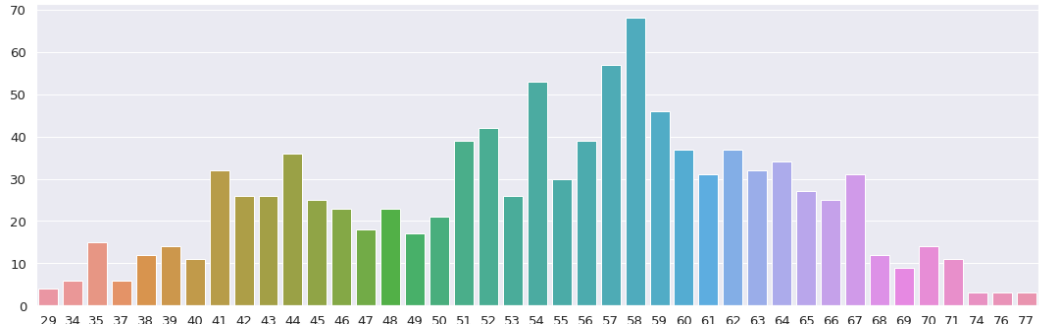


**Рис. 5. Кількість пацієнтів в різному віці**

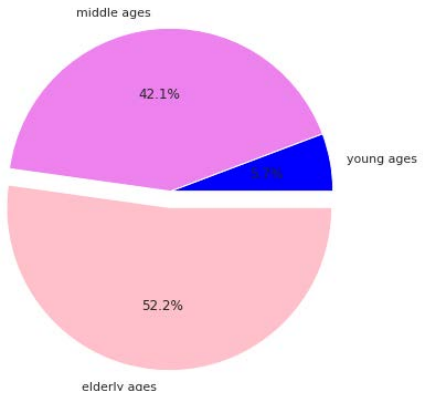


**Рис. 6. Різні вікові категорії**

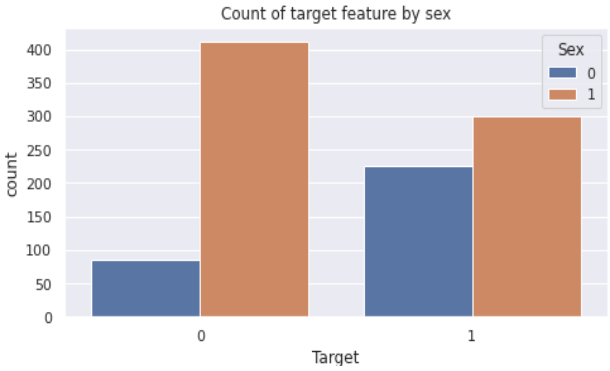


**Рис. 7. Порівняння хворих за статтю**

Наші дані не мають нульових значень і це хороший показник для майбутніх досліджень.

За допомогою методу df.describe() ми можемо побачити опис кожного стовпця, а саме кількість непорожніх значень, середнє значення, стандартне відхилення, квартилі та максимальні значення.

Далі перевіримо зв'язок кожної змінної за допомогою кореляції цільової змінної.

Чотири ознаки («CP», «Restbps», «MaxHR», «Slope») позитивно корелюють з цільовою ознакою.

Побудуємо діаграму, де зможемо побачити кількість хворих та здорових людей.

Далі проведемо аналіз окремих змінних.

Дана діаграма показу, що найбільше кількість пацієнтів, яким 58 років.

Тепер поділимо наш вік на різні категорії.

Тут ми бачимо, що люди похилого віку найбільше страждають від хвороб серця, а молодь – найменше.

Проведемо багатофакторний аналіз, порівнявши кількість здорових і не здорових людей за статтю.

Звідси випливає, що найбільше хворіють саме чоловіки.

Наступний графік показує залежність віку і значення максимального тиску пацієнта, а також чи хворий пацієнт з таким співвідношенням.


**Рис. 8. Співвідношення віку та максимального тиску**

Більшість пацієнтів, які мають високий тиск, виявлено серцево-судинну хворобу, незалежно від віку.

**Розробка та навчання моделі для класифікації**

Проаналізувавши вхідні дані, перейдемо до моделі прогнозування міток, базуючись на вхідних даних. В першу чергу ми імпортуємо наші дані в конструктор машинного навчання Azure. Далі потрібно провести певні перетворення даних для поліпшення навчання моделі. Наші дані мають достатньо категоріальних стовпців, тому використовуючи модель Edit Metadata (Редагування метаданих) позначимо ці стовпці.

Далі ці стовпці перетворимо у ряд бінарних індикаторів. Для цього використовуємо модуль Convert to Indicator Values (Перетворення на значення індикаторів). На виході наші дані будуть мати наступний вигляд (рис. 10).

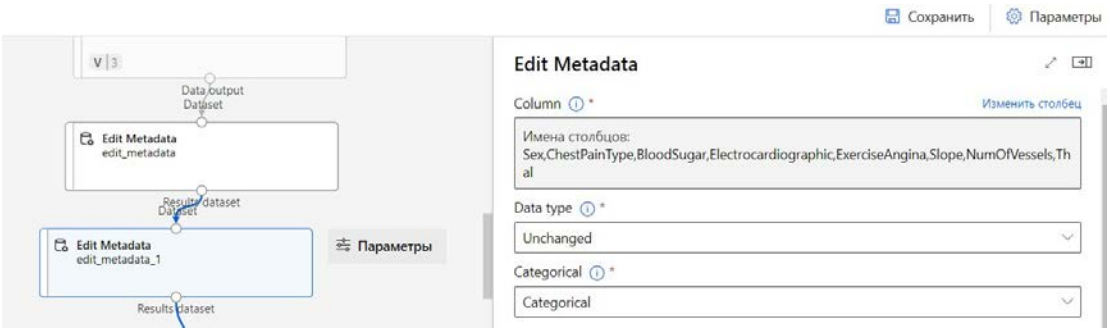


**Рис. 9. Співвідношення віку та максимального тиску**

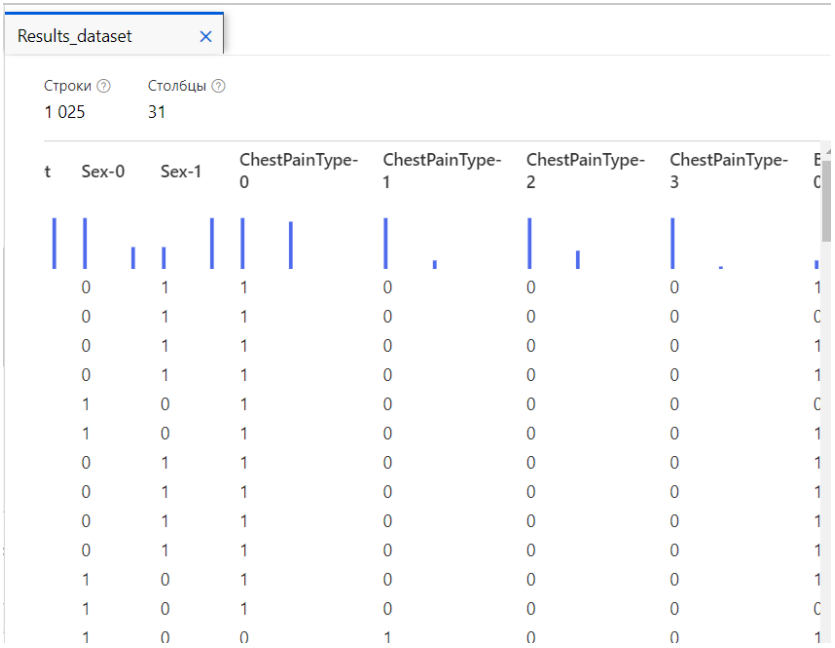


**Рис. 10. Перетворення стовпців**

Тепер на прикладі стовпця «Sex» ми бачимо, що кожна стать це окремий стовпець. Надалі буде легше оцінити важливість кожного значення.

Останній крок перед навчанням це поділ наших даних на навчальні та тестові набори даних, щоб оцінити, чи правильно працює модель машинного навчання. Для поля модуля «Частка рядків у першому вихідному наборі даних» задаємо значення 0.75, що означає поділ даних у співвідношенні 80 на 20. Тобто 20 відсотків піде на тестування нашої моделі. Також важливо поставити значення «true» для поля «Випадковий поділ». Це бажаний варіант, коли ви створюєте навчальні та тестові набори даних.

Далі підключаємо наш алгоритм, а саме модуль «Two-Class Boosted Decision Tree component» ( Посилене дерево рішень). Цей модуль не просто будує дерева рішень, і обирає краще, але й виправляє помилки попередніх дерев. Прогнози ґрунтуються на сукупності всіх дерев, які складають прогноз.

Також важливо налаштувати параметри навчання. Для кращого прогнозу буде використано модуль «Tune Model Hyperparameters» (Налаштування гіперпараметрів моделі). Для кращого розуміння, розберемось з такими поняттями:

Параметри моделі – це параметри в моделі, які необхідно визначити за допомогою набору навчальних даних. Це встановлені параметри. Наприклад: кількість точок поділу в дереві рішень тощо.

Гіперпараметри – це налаштовувані параметри, які дозволяють контролювати процес навчання моделі. Наприклад, у випадку нейронних мереж ви встановлюєте кількість прихованих шарів та число вузлів у кожному з них. Ефективність моделі визначається гіперпараметрами.

Налаштування гіперпараметрів – це процес пошуку конфігурації гіперпараметрів, що забезпечує найкращу продуктивність. Цей процес потребує значних обчислювальних ресурсів та вимагає виконання великого обсягу ручної роботи. Це досягається шляхом навчання кількох моделей з використанням одного і того ж алгоритму та даних навчання, але з різними значеннями гіперпараметрів. Результуюча модель кожного тренувального прогону потім оцінюється, щоб визначити показник продуктивності, для якого потрібно оптимізувати (наприклад, точність), і вибирається найбільш продуктивна модель.

Також до нашої моделі додаємо модуль «Permutation Feature Importance» (Важливість функції перестановки). Даний модуль допоможе оцінили важливість кожної функції методом перестановки.

Після навчання моделі важливо оцінити її продуктивність. Існує безліч показників продуктивності та методології для оцінки того, наскільки добре модель робить прогнози. Для цього підключаємо модуль «Score Model» та «Evaluate Model». Перший модель показує прогнозовану мітку для кожного рядка тестових даних, в той час як другий допомагає оцінити якість моделі за допомогою метрик, які розглянемо в наступному розділі.

Наша сконструйована модель виглядає наступним чином (рис. 11).

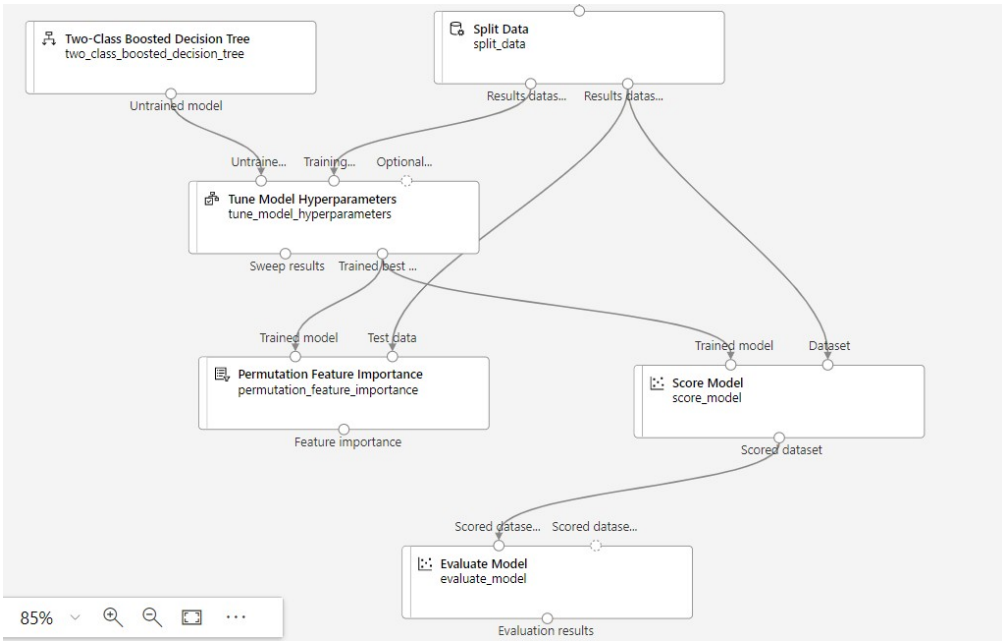


**Рис. 11. Сконструйована модель**

***Оцінка та аналіз якості моделі***

Розробка моделі зазвичай поділяється на три етапи. Спочатку модель навчається на навчальному наборі даних, що підходить для даної задачі. По-друге, під час навчання модель постійно перевіряється на даних, які є частиною навчальних даних, щоб оцінити продуктивність моделі на невидимих даних. Нарешті після того, як модель закінчила навчання, вона тестується на тестовому наборі даних, для якого повинні бути розраховані остаточні метрики. Незалежно від того, яка метрика використовується, вона може бути такою ж інформативною, як і продуктивність класифікатора на тестових даних.

Існує чимало показників з метою оцінки моделей машинного навчання у різних додатках. Більшість їх можна розділити на дві категорії залежно від типів прогнозів у моделях машинного навчання.

Ось деякі з популярних метрик класифікації, які ми збираємося розглянути:

– Акуратність:

– Точність:

– Повнота;

– Оцінка F1.

Матриця плутанини представляє собою ключовий елемент, що використовується для оцінки продуктивності моделі класифікації у машинному навчанні, хоча сама по собі не є метрикою. Сутність полягає у створенні двовимірної таблиці, що відображає фактичні та передбачені значення. В даному випадку виникає потреба розробити класифікатор, який здатен визначати пацієнтів на хворих і здорових.

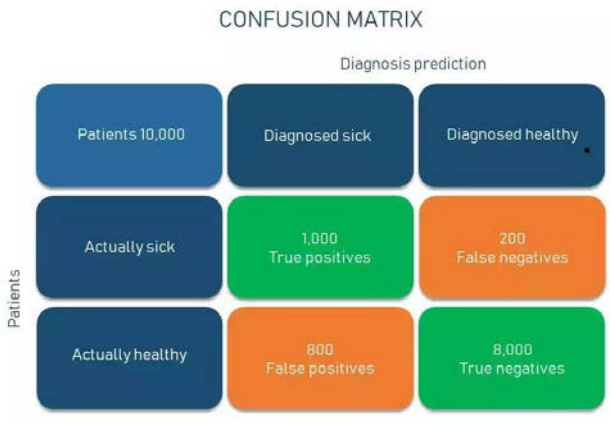


**Рис. 12. Схема матриці плутанини**

Істинно-позитивно (TP) – клас спрогнозавано вірно і вірно в дійсності (пацієнти, які хворі і хворіють діагнозом);

Істинно-негативно (TN) – клас спрогнозовано хибно і хибно в дійсності (пацієнти, які здорові і діагностовані здоровими);

Хибно-позитивні (FP) – клас спрогнозовано як істинний, але в дійсності хібний (пацієнти, які здорові, але діагностовані як хворі); і

Хибно-негативні (FN) – клас спрогнозовано як хибні, але насправді правдивий (пацієнти, які хворі, але діагностовані здоровими).

Розібравшись з деталями схеми, переглянемо нашу отриману матрицю плутанини (рис. 13).


**Рис. 13. Матриці плутанини**

По даній матриці видно, що майже усі тестові дані спрогнозовано вірно. Акуратність використовується до розрахунку частки правильних прогнозів від загального числа. Це кількість правильних прогнозів, поділена на загальну кількість прогнозів.

Формула виглядає наступним чином:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Будучи одним із найпоширеніших показників класифікації, точність дуже інтуїтивно зрозуміла і проста для розуміння та реалізації: вона варіюється від 0 до 100 відсотків або від 0 до 1. Якщо ми візьмемо модель діагнозу «здоровий/хворий», з усіх 10 000 пацієнтів модель правильно класифікує 9 000 пацієнтів, або 90 відсотків, або 0,9, якщо ми вимірюємо від 0 до 1. Отже, наш показник акуратності дорівнює 0.99.

Точність показує, яка частка із усіх позитивних прогнозів була правильною. Щоб розрахувати його, ви ділите кількість правильних позитивних результатів (TP) на загальну кількість всіх позитивних результатів (TP + FP), передбачених класифікатором.

Формула виглядає наступним чином:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Повнота показує частку правильних позитивних прогнозів від усіх позитивних результатів, які б зробила модель. Щоб обчислити його, ви ділите всі справжні позитивні результати у сумі всіх справжніх позитивних і хибних негативних результатів у наборі даних. Таким чином, повнота

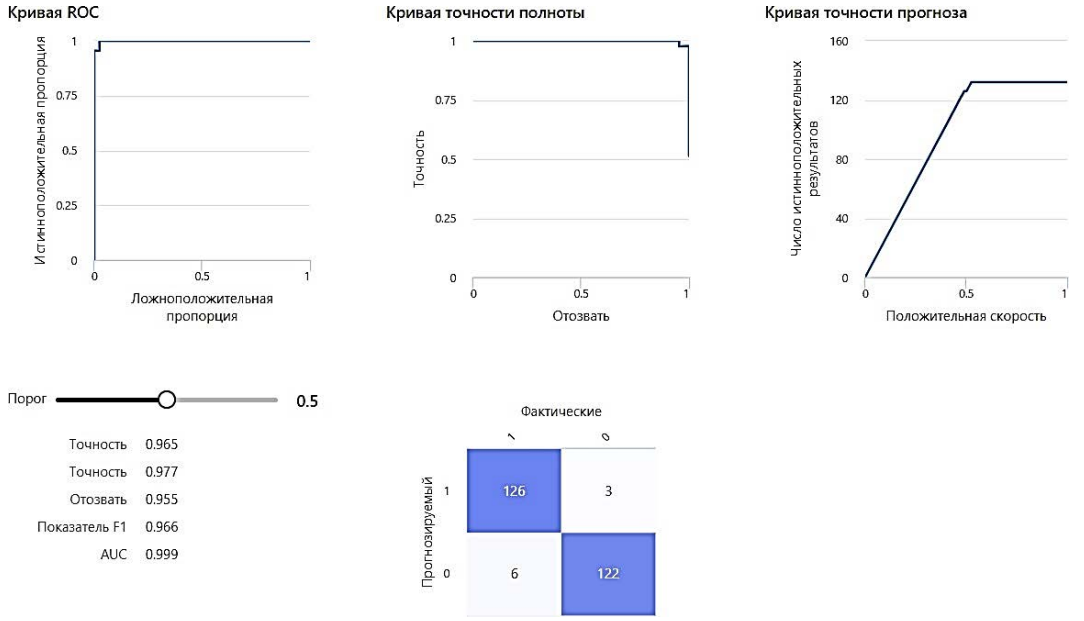


**Рис. 14. Метрики навченої моделі**

вказує на пропущені позитивні прогнози на відміну від метрики точності, яку ми пояснили вище.

Формула виглядає наступним чином:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Оцінка F1 намагається знайти баланс між точністю та повноти, обчислюючи їхнє середнє гармонійне. Це міра точності тесту, де максимально можливе значення дорівнює 1. Це вказує на ідеальну точність та повноту.

Формула виглядає наступним чином:

$$F_1 = \left( \frac{\text{recall}^{-1} + \text{precision}^{-1}}{2} \right)^{-1} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \quad (4)$$

Усі метрики нашої моделі можемо переглянути на рисунку 14.

Дослідимо ще одну з найбільш важливих метрик для перевірки якості моделі класифікації, на яку ми будемо найбільше опиратись – AUC-ROC крива (рис. 14).

Крива AUC-ROC є вимірюванням продуктивності для завдань класифікації при різних порогових значеннях. ROC являє собою криву ймовірності, а AUC є ступенем або мірою роздільності. Він каже, наскільки модель здатна розрізняти класи. Чим вище AUC, тим краще модель передбачає 0 класів як 0 і 1 клас як 1. За аналогією, чим вище AUC, тим краще модель розрізняє пацієнтів із захворюванням і без захворювання.

Проаналізувавши кожну метрику і отримавши хорошу оцінку AUC, можна впевнено сказати, що наша модель має гарні результати та готова до тесту у режимі реального часу.

**Висновки.** У статті було вирішена актуальна задача класифікації медичних даних на основі конструктора машинного навчання Azure. Був обраний метод контрольованого машинного навчання, а саме метод класифікації, для вирішення поставленої задачі. Для вирішення проблеми прогнозування серцевих захворювань було обрано хмарну платформу Microsoft Azure з сервісом Azure Machine Learning. Розроблена модель прогнозування серцевих захворювань включає модулі: набір даних, два модулі редагування даних, вибір стовпців для навчання, виконання R скрипта, поділ даних, двокласове збільшуване дерево прийняття рішень, налаштування гіперпараметрів моделі, навчання моделі, оцінка та аналіз моделі. Наступним етапом було оцінювання якості моделі за допомогою метрик: акуратність, точність, повнота, оцінка F1 та AUC-ROC крива.

**Список літератури:**

1. Puleko I.V., Poberezhna M.O. Solution of the Classification Problem for Medical Data on the Basis of Azure Machine Learning Service and the Developed Web Application. *Тези доповідей IV Всеукраїнської науково-технічної конференції: «Комп'ютерні технології: інновації, проблеми, рішення*, 18-20 листопада 2021 року. Житомир: «Житомирська політехніка, 2021. С. 13-14.

2. Гадецька З., Меркотан М. Аналіз і прогнозування рівня захворюваності на Covid-19 в країнах європейського регіону. *Економіка та суспільство*. 2022. Вип. 39. https://doi.org/10.32782/2524-0072/2022-39-20.

3. Скопівський С.Я. Аналіз методів прогнозування інфекційних захворювань. *Вісник Хмельницького національного університету*. 2022. Вип. 4 (311). С. 237-240.

4. Гойко О.В. Сучасні технології обробки й аналізу медичних даних. *Медична інформатика та інженерія*. 2009. Вип. 4. С. 39-44.

5. Івченко В.К., Івченко А.В., Гальченко В.Я., Івченко Д.В., Швець О.І. Прогнозування результатів лікування переломів довгих кісток у хворих на цукровий діабет засобами інтелектуального та статистичного аналізу даних. *Журнал «Травма».* 2013. Том 14, № 4. URL: http://www.mif-ua.com/archive/article/3680

6. Півошенко В.В., Кулик М.С., Іванов Ю.Ю., Васюра А.С. Аналіз та експериментальне дослідження методу безмодельного навчання з підкріпленням. *Вісник Вінницького політехнічного інституту*. 2019. Вип. 3. С. 40-49.

**Puleko I.V., Poberezhna M.O., Kravchenko S.M., Dmytrenko I.A. SOLUTION OF THE CLASSIFICATION PROBLEM FOR MEDICAL DATA ON THE BASIS OF AZURE MACHINE LEARNING SERVICE AND THE DEVELOPED WEB APPLICATION**

*At the moment, machine learning (ML) is one of the most relevant directions in software development. The healthcare industry is especially widely using ML to create web applications that are able to predict risks and detect diseases in the early stages, in order to improve patient care. The process of developing such a web application described in the work includes three stages: creating a machine learning model in Azure, developing the web application itself and integrating the developed model into it, as well as deploying the web service using Azure. For the first stage, Azure Machine Learning Designer was used, which provides a graphical environment for creating ML models. The article describes in detail the steps of creating a model, such as creating a pipeline, importing data, preparing data, training the model using the chosen algorithm, as well as evaluating and analyzing the resulting model. The development uses a decision tree model, which has a tree-like structure: internal nodes represent data characteristics, branches reflect decision-making rules, and end nodes represent final results. This model is well suited for solving classification problems because the target variable is discrete. After the successful creation and testing of the model, the transition to the next stage was implemented - the creation of a web application using Vue.js, CSS and Node.js technologies. The main purpose of this application is to enable doctors to quickly diagnose patients and other users to access test results. For this, an interface has been created that will allow you to enter input data and receive a predicted result. The next implemented stage is the deployment of machine learning models. This means bringing the models into a production environment where they can be used in real time. After the model was deployed, its performance was tested using test data in JSON format to ensure that the web service returned the expected results.*

***Key words:*** *data analysis, classification model, machine learning, artificial intelligence, programming technologies, software.*